Objectives

- SLogo Planning

SLogo Information

- On Final Project page, for each team
  - Statistics
  - Documentation

Today's Goals

- Planning: Design and Analysis
  - What are the components that need to be completed?
  - How should we complete those components?
- More code iteration
  - Make sure you understand the code base
- More team collaboration

You have all the skills needed to understand this code base

What Steps Need To Be Completed?

- By the Team
- Process to figure out what needs to be completed

What Steps Need To Be Completed?

- Model: Turtle
  - API
  - State
- GUI
  - Canvas – displays turtle
  - Command interface
  - Listeners
  - Multiple workspaces
  - Turtle info displayed (toggable)
  - More options/buttons (optional)
- **TESTING!**
- Parsing SLogo language
  - Handle instructions
  - Handle *aliases*
  - **Handle errors** appropriately
- Evaluating expressions
  - Manipulate turtle
- File handling
  - Read file of SLogo files
  - Save SLogo commands

Dependencies?

Dependencies

- Interpreter classes (tokens, analyzer, expression) are very dependent on each other
- Need to hook GUI to Interpreter
- Need to hook Turtle to GUI and Interpreter

- Can test without other pieces
  - Good first step
  - But, easier and more satisfying to see results displayed

Effect of Extensions

- Extensions could affect your code design
  - Where could change → abstraction

- Decision?
  - May change your minds after start working on the code
    - Deadline: Wed, Dec 7
  - Top vote getters

Plan

- Tasks/Steps
  - Testing
  - Think about iterative development
    - Wednesday deadline: FD 50 working at least
- Division of tasks
  - # of people per part
- Deadlines

Goals

- Implement one instruction completely
  - Involves a lot of different pieces
- Don't go too far in breadth, more depth
  - See design issues sooner
    - "We need method/functionality X in class Y"

Secondary Goals

- You're going to figure out that your design isn't perfect--maybe not even good!
  - Fix smaller and/or more critical things
    - Refactoring!
  - Note larger things
    - analysis/post-mortem due at end of finals week

Good judgment comes from experience.
How do you get experience?
Bad judgment works every time.

TODO

- Recommendations
  - Review the Javadocs – abstraction of the code
    - Add more comments as you understand pieces of the code.
  - Trace through the code
    - Two starts: SLogoInterpreter and TurtleField
  - Break into small pieces
  - Draw pictures
  - Leverage the team
- Lots of thinking before implementation
  - How should components work together?
- Preliminary Implementation: Wed, November 30

Design Questions

- How do you want the GUI to look?
- How will the commands manipulate the turtle?
- How will you handle aliases?